MT461/MT5461
Theory of Error Correcting Codes

Mark Wildon, mark.wildon@rhul.ac.uk

The extra content on the syllabus for MT5461 is on Reed–Solomon codes and cyclic codes over finite fields. These codes are examples of the linear codes that will be covered in Part C of the main lectures.

Definition 1.1

A *field* is a set of elements $\mathbf{F}$ with two operations, $+$ (addition) and $\times$ (multiplication), and two special elements $0, 1 \in \mathbf{F}$ such that $0 \neq 1$ and

(1) $a + b = b + a$ for all $a, b \in \mathbf{F}$;

(2) $0 + a = a + 0 = a$ for all $a \in \mathbf{F}$;

(3) for all $a \in \mathbf{F}$ there exists $b \in \mathbf{F}$ such that $a + b = 0$;

(4) $a + (b + c) = (a + b) + c$ for all $a, b, c \in \mathbf{F}$;

(5) $a \times b = b \times a$ for all $a, b \in \mathbf{F}$;

(6) $1 \times a = a \times 1 = a$ for all $a \in \mathbf{F}$;

(7) for all non-zero $a \in \mathbf{F}$ there exists $b \in \mathbf{F}$ such that $a \times b = 1$;

(8) $a \times (b \times c) = (a \times b) \times c$ for all $a, b, c \in \mathbf{F}$;

(9) $a \times (b + c) = a \times b + a \times c$ for all $a, b, c \in \mathbf{F}$.

Definition 1.2

The *order* of a finite field $\mathbf{F}$ is defined to be the number of elements in $\mathbf{F}$.

Exercise: show from the axioms for a field that if $\mathbf{F}$ is a field then $a \times 0 = 0$ for all $a \in \mathbf{F}$. Show that if $x \in \mathbf{F}$ then x has a unique additive inverse, and that if $x \neq 0$ then x has a unique multiplicative inverse.

Exercise: show from the axioms for a field that if $\mathbf{F}$ is a field and a, $b \in \mathbf{F}$ are such that $a \times b = 0$, then either $a = 0$ or $b = 0$.

Theorem 1.3

Let p be a prime. The set $\mathbf{F}_p = \{0, 1, \ldots, p-1\}$ with addition and multiplication defined modulo p is a field.

Lemma 1.5 (Division algorithm)

Let $\mathbf{F}$ *be a field, let* $f(x) \in \mathbf{F}[x]$ *be a non-zero polynomial and let* $g(x) \in \mathbf{F}[x]$*. There exist polynomials* $s(x), r(x) \in \mathbf{F}[x]$ *such that*

$$g(x) = s(x)f(x) + r(x)$$

and either $r(x) = 0$ *or* $\deg r(x) < \deg f(x)$*.*

Exercise: Let $f(x) = x^3 + x + 1 \in \mathbf{F}_2[x]$. Find the quotient and remainder when $g(x) = x^5 + x^2 + x$ is divided by $f(x)$.

Other Results on Polynomials

For Reed–Solomon codes we shall need the following properties of polynomials.

Lemma 1.6

Let $\mathbf{F}$ *be a field.*

(i) *If* $f(x) \in \mathbf{F}[x]$ *has* $a \in \mathbf{F}$ *as a root, i.e.* $f(a) = 0$*, then there is a polynomial* $g(x) \in \mathbf{F}[x]$ *such that* $f(x) = (x - a)g(x)$*.*

(ii) *If* $f(x) \in \mathbf{F}[x]$ *has degree* d *then* $f(x)$ *has at most* d *distinct roots in* $\mathbf{F}$*.*

(iii) *If* $f, g \in \mathbf{F}[x]$ *both have degree* $< n$ *and there exist distinct* $a_1, \ldots, a_n \in \mathbf{F}$ *such that* $f(a_i) = g(a_i)$ *for each* $i \in \{1, \ldots, n\}$ *then* $f(x) = g(x)$*.*

Polynomial Interpolation

Lemma 1.7 (Polynomial interpolation)

Let $\mathbf{F}$ *be a field. Let*

$$a_1, a_2, \ldots, a_k \in \mathbf{F}$$

be distinct and let $y_1, y_2, \ldots, y_k \in \mathbf{F}$*. The unique polynomial* $f(x) \in \mathbf{F}[x]$ *of degree* $< k$ *such that* $f(a_i) = y_i$ *for all* i *is*

$$f(x) = \sum_{i=1}^{n} y_i \frac{\prod_{j \neq i}(x - a_j)}{\prod_{j \neq i}(a_i - a_j)}.$$

Part 1: Reed Solomon codes

§2 Definition and basic properties of Reed–Solomon codes

Definition 2.1

Let p be a prime and let k, $n \in \mathbf{N}$ be such that $k \leq n \leq p$. Let

$$a_1, a_2, \ldots, a_n$$

be distinct elements of $\mathbf{F}_p$. For each polynomial $f(x) \in \mathbf{F}_p[x]$ we define a word $u(f) \in \mathbf{F}_p^n$ by

$$u(f) = (f(a_1), f(a_2), \ldots, f(a_n)).$$

The *Reed–Solomon code* associated to the parameters p, n, k and the field elements $a_1, a_2, \ldots, a_n$ is the length n code over $\mathbf{F}_p$ with codewords

$$\{u(f) : f \in \mathbf{F}_p[x],\ \deg f \leq k-1\}.$$

Example 2.2

Let $p = 5$ and let $k = 2$.

(1) If $n = 3$ and we take $a_1 = 0$, $a_2 = 1$ and $a_3 = 2$, then the associated Reed–Solomon code has a codeword

$$(f(0), f(1), f(2))$$

for each $f(x) \in \mathbf{F}_p[x]$ of degree ≤ 1. If $f(x) = bx + c$ then

$$u(f) = (c, b + c, 2b + c)$$

so the full set of codewords is

$$\{(c, b + c, 2b + c) : b, c \in \mathbf{F}_5\}.$$

Exercise: show that this code is 1-error detecting, but not 2-error detecting.

(2) If $n = 4$ and we take a_1, a_2, a_3 as before, and $a_4 = 3$ then we get an extension of the code in (1).

Exercise: Show that if $C = \{(c, b+c, 2b+c, 3b+c) : b, c \in \mathbf{F}_5\}$ then C is 2-error detecting and 1-error correcting. (*Hint:* Question 4 on Sheet 1 will help, particularly for the latter part.)

Basic properties

For the rest of this section, fix parameters p, n, k and field elements $a_1, a_2, \ldots, a_n$. Let $RS_{p,n,k}$ denote the associated Reed–Solomon code over $\mathbf{F}_p$.

Lemma 2.3

The Reed–Solomon code $RS_{p,n,k}$ has size p^k.

The next lemma gives a lower bound on the Hamming distances between codewords in the Reed–Solomon code.

Lemma 2.4

If f, $g \in \mathbf{F}_p[x]$ are distinct polynomials of degree $\leq k-1$ then

$$d(u(f), u(g)) \geq n - k + 1.$$

Theorem 2.5

The minimum distance of $RS_{p,n,k}$ is $n - k + 1$.

Remarks on Lemma 2.4 and Theorem 2.5

(1) Suppose that $f, g \in \mathbf{F}_p[x]$ are polynomials of degree $< k$. Then by Lemma 2.4, $d(u(f), u(g)) \geq n - k + 1$. In particular $u(f) \neq u(g)$. This gives another proof that the Reed–Solomon code $RS_{p,n,k}$ has size p^k.

(2) The interpolating polynomial given by Lemma 1.7 is unique. Proof was omitted in Lecture 2, so will give now.

Corollary 2.6

Let p be a prime. If k, $e \in \mathbf{N}$ are such that $k + 2e \leq p$ then the Reed–Solomon code $RS_{p,k+2e,k}$ is e-error correcting.

Optimality of Reed Solomon codes

The Singleton Bound (to be proved in Part B of the main course) states that any p-ary code of length n and minimum distance d has at most p^{n-d+1} codewords. By Lemma 2.3 and Theorem 2.5, the Reed–Solomon codes meet this bound, and so have the largest possible size for their length and minimum distance.

Example 2.7

Suppose we use the Reed–Solomon code with $p = 5$, $n = 4$ and $k = 2$ evaluating at $a_1 = 0$, $a_2 = 1$, $a_3 = 2$, $a_4 = 3$, as in Example 2.2(2). By Corollary 2.6, this code is 1-error correcting. Suppose we receive $v = (4, 0, 3, 0)$.

Given any two positions i and j, it follows from Lemma 1.7 that there is a unique polynomial g of degree < 2 such that $g(a_i) = v_i$ and $g(a_j) = v_j$.

The table on the next slide shows the interpolating polynomials for each pair of positions and the corresponding codewords. For example, to find $f(x)$ such that $f(0) = 4$ and $f(2) = 3$, we use Lemma 1.7 and get

$$f(x) = 4\,\frac{x-2}{0-2} + 3\,\frac{x-0}{2-0} = 3(x-2) - x = 2x + 4.$$

Conditions on f	Solution	Codeword $u(f)$
$f(0) = 4,\ f(1) = 0$	$f(x) = 4 + x$	$(4, 0, 1, 2)$
$f(0) = 4,\ f(2) = 3$	$f(x) = 4 + 2x$	$(4, 1, 3, 0)$
$f(1) = 0,\ f(2) = 3$	$f(x) = 2 + 3x$	$(2, 0, 3, 1)$
$f(0) = 4,\ f(3) = 0$	$f(x) = 4 + 2x$	$(4, 1, 3, 0)$
$f(1) = 0,\ f(3) = 0$	$f(x) = 0$	$(0, 0, 0, 0)$
$f(2) = 3,\ f(3) = 0$	$f(x) = 4 + 2x$	$(4, 1, 3, 0)$

In practice, we would stop as soon as we found the codeword $(4, 1, 3, 0)$ since $d(4130, 4030) = 1$, and by the exercise on page 19 of the main lecture notes, there is at most one codeword within distance 1 of any given word.

§3 Efficient Decoding of Reed–Solomon codes

As usual we work with the Reed–Solomon code $RS_{p,n,k}$ where p is prime and n, $k \in \mathbf{N}$, and polynomials are evaluated at $a_1, a_2, \ldots, a_n$. Assume that $n = k + 2e$, so by Corollary 2.6 the code is e-error correcting.

Theorem 3.1 (Key Equation)

Suppose that the codeword

$$u(f) = (f(a_1), \ldots, f(a_n))$$

is transmitted and the word $(v_1, \ldots, v_n)$ *is received. If there are* $\leq e$ *errors in transmission then there exist polynomials*

- $Q(x)$ *of degree* $\leq k + e - 1$
- $E(x)$ *of degree* $\leq e$*,*

such that the Key Equation

$$Q(a_i) = v_i E(a_i)$$

Using Key Equation to Decode

It is not at all obvious why the Key Equation is helpful. We first show that any solution to it can be used to decode a received word.

Lemma 3.2

Suppose that the codeword

$$u(f) = (f(a_1), \ldots, f(a_n))$$

is transmitted and the word $(v_1, \ldots, v_n)$ *is received. If* $E(x)$ *and* $Q(x)$ *satisfy the Key Equation, and the number of errors in transmission is* $\leq e$*, then* $Q(x) = f(x)E(x)$ *and so* $f(x) = Q(x)/E(x)$*.*

Solving Key Equation

Lemma 3.3

Suppose that the word $(v_1, \ldots, v_n)$ is received. The polynomials

$$Q(x) = Q_0 + Q_1 x + \cdots + Q_{k+e-1} x^{k+e-1}$$
$$E(x) = E_0 + E_1 x + \cdots + E_e x^e$$

in $\mathbf{F}_p[x]$ satisfy the Key Equation if and only if

$$Q_0 + a_i Q_1 + a_i^2 Q_2 + \cdots + a_i^{k+e-1} Q_{k+e-1} = v_i (E_0 + a_i E_1 + a_i^2 E_2 + \cdots + a_i^e E_e)$$

for each $i \in \{1, \ldots, n\}$.

Lemma 3.3 [continued]

An equivalent condition is that

$$\begin{pmatrix} 1 & a_1 & a_1^2 & \cdots & a_1^{k+e-1} & -v_1 & -v_1 a_1 & \cdots & -v_1 a_1^e \\ 1 & a_2 & a_2^2 & \cdots & a_2^{k+e-1} & -v_2 & -v_2 a_2 & \cdots & -v_2 a_2^e \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & a_n & a_n^2 & \cdots & a_n^{k+e-1} & -v_n & -v_n a_n & \cdots & -v_n a_n^e \end{pmatrix} \begin{pmatrix} Q_0 \\ Q_1 \\ \vdots \\ Q_{k+e-1} \\ E_0 \\ E_1 \\ \vdots \\ E_e \end{pmatrix}$$

Example 3.4

Let $p = 5$, let $k = 2$, let $e = 1$ (so $n = 4$) and let $a_1 = 0$, $a_2 = 1$, $a_3 = 2$, $a_4 = 3$. With these parameters, the Key Equation for the polynomials $Q(x) = Q_0 + Q_1 x + Q_2 x^2$ and $E(x) = E_0 + E_1 x$ is

$$\begin{pmatrix} 1 & 0 & 0^2 & -v_1 & 0 \\ 1 & 1 & 1^2 & -v_2 & -v_2 \\ 1 & 2 & 2^2 & -v_3 & -2v_3 \\ 1 & 3 & 3^2 & -v_4 & -3v_4 \end{pmatrix} \begin{pmatrix} Q_0 \\ Q_1 \\ Q_2 \\ E_0 \\ E_1 \end{pmatrix} = 0.$$

or equivalently

$$\begin{pmatrix} 1 & 0 & 0 & 4v_1 & 0 \\ 1 & 1 & 1 & 4v_2 & 4v_2 \\ 1 & 2 & 4 & 4v_3 & 3v_3 \\ 1 & 3 & 4 & 4v_4 & 2v_4 \end{pmatrix} \begin{pmatrix} Q_0 \\ Q_1 \\ Q_2 \\ E_0 \\ E_1 \end{pmatrix} = 0.$$

(1) Suppose we receive the word 4130. (This is the codeword for $f(x) = 4 + 2x$. Then $v_1 = 4$, $v_2 = 1$, $v_3 = 3$, $v_4 = 0$ and we must solve

$$\begin{pmatrix} 1 & 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 4 & 4 \\ 1 & 2 & 4 & 2 & 4 \\ 1 & 3 & 4 & 0 & 0 \end{pmatrix} \begin{pmatrix} Q_0 \\ Q_1 \\ Q_2 \\ E_0 \\ E_1 \end{pmatrix} = 0.$$

The kernel is two dimensional, spanned by the vectors

$$(0, 4, 2, 0, 1)^t, \quad (4, 2, 0, 1, 0)^t.$$

The first vector gives $Q(x) = 4x + 2x^2$ and $E(x) = x$, so we decode using $f(x) = Q(x)/E(x) = 4 + 2x$ to get $u(f) = 4130$.

(2) Suppose we receive the word 4030. Then $v_1 = 4$, $v_2 = 0$, $v_3 = 3$, $v_4 = 0$ and we must solve

$$\begin{pmatrix} 1 & 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 \\ 1 & 2 & 4 & 2 & 4 \\ 1 & 3 & 4 & 0 & 0 \end{pmatrix} \begin{pmatrix} Q_0 \\ Q_1 \\ Q_2 \\ E_0 \\ E_1 \end{pmatrix} = 0.$$

The kernel is one dimension spanned by $(1, 2, 2, 4, 1)^t$. So we take $Q(x) = 1 + 2x + 2x^2$ and $E(x) = 4 + x$. Polynomial division gives

$$Q(x)/E(x) = 2x + 4$$

so we decode using $f(x) = 2x + 4$ to get $u(f) = 4130$.

(3) Finally suppose we receive 4020. Then the kernel is one dimensional, spanned by $(4, 3, 3, 1, 0)$. So we take $Q(x) = 4 + 3x + 3x^2$ and $E(x) = 1$, but $Q(x)/E(x)$ does not have degree ≤ 1, so we are unable to decode. Since the Key Equation method always works when $\leq e$ errors occur, we know that ≥ 2 errors have occurred, but we are unable to correct them.

§2 Cyclic codes

Cyclic codes are a special type of linear code. In Part C of the main course we will consider linear codes over the binary alphabet. Here we work more generally over a finite field $\mathbf{F}_p$ of prime order.

Definition 4.1

Let p be prime. A code C over $\mathbf{F}$ is *linear* if

(i) for all $u \in C$ and $a \in F$ we have $au \in C$;

(ii) for all u, $v \in C$ we have $u + v \in C$.

Definition 4.2

Let p be a prime. A code C over $\mathbf{F}_p$. is said to be *cyclic* if C is linear and

$$(u_0, u_1, \ldots, u_{n-1}) \implies (u_{n-1}, u_0, \ldots, u_{n-2}) \in C.$$

Examples of Cyclic Codes

Example 4.3

(1) Let p be prime. The repetition code of length n over $\mathbf{F}_p$ is cyclic.

(2) Let C be all binary words of length $n \in \mathbf{N}$ with evenly many 1s. We may define C using addition in $\mathbf{F}_2$ by

$$C = \{(u_0, \ldots, u_{n-1}) : u_i \in \mathbf{F}_2, u_0 + \ldots + u_{n-1} = 0\}.$$

Then C is a cyclic code.

(3) Let D be the binary code $\{0000, 1010, 0101, 1111\}$. *Exercise:* check that D is linear. The shift map acts on D by fixing 0000 and 1010 and swapping 1010 and 0101, so D is cyclic.

Correspondence with polynomials

Definition 4.4

Let p be prime. Given a codeword

$$u = (u_0, u_1, \ldots, u_{n-1}) \in \mathbf{F}_p^n.$$

we define the *polynomial corresponding to u* to be

$$u_0 + u_1 x + \cdots + u_{n-1} x^{n-1}$$

and write

$$u \longleftrightarrow u_0 + u_1 x + \cdots + u_{n-1} x^{n-1}.$$

Definition 4.5

Let p be prime. The ring $\mathbf{F}_p[x]/(x^n-1)$, read as '$\mathbf{F}_p[x]$ modulo x^n-1' has elements all polynomials in $\mathbf{F}_p[x]$ of degree $< n$. Given

$$f(x) = a_0 + a_1x + \cdots + a_{n-1}x^{n-1}$$
$$g(x) = b_0 + b_1x + \cdots + b_{n-1}x^{n-1}$$

in $\mathbf{F}_p[x]/(x^n-1)$ we define their sum, in the obvious way, to be

$$f(x)+g(x) = (a_0+b_0)+(a_1+b_1)x+\cdots+(a_{n-1}+b_{n-1})x^{n-1}.$$

The product $f(x)g(x) \in \mathbf{F}_p[x]/(x^n-1)$ is defined by taking the normal product $f(x)g(x) \in \mathbf{F}_p[x]$ and then taking the remainder on division by x^n-1.

Remarks 4.6

(1) This definition is analogous to the earlier definition (see Theorem 1.3) of the finite field $\mathbf{F}_p$ as the set $\{0, 1, \ldots, p-1\}$ with addition and multiplication defined by performing these operations in $\mathbf{Z}$, and then taking the remainder after division by p.

(2) We will assume that $\mathbf{F}[x]/(x^n - 1)$ is a ring, i.e. it satisfies all the axioms, except (7), on page 2. It is routine but time-consuming to check they all hold.

This result also follow from the general theory of quotient rings. Defined this way, $\mathbf{F}[x]/(x^n - 1)$ is the set of cosets

$$f(x) + \langle (x^n - 1) \rangle$$

of the ideal in $\mathbf{F}[x]$ generated by $(x^n - 1)$. Our definition makes a specific choice of coset representatives.

Cyclic shifts correspond to multiplication by x

Lemma 4.7

Let p be a prime and let $u = (u_0, u_1, \ldots, u_{n-1}) \in \mathbf{F}_p^n$. Let

$$f(x) = u_0 + u_1 x + \cdots + u_{n-1}x^{n-1} \in \mathbf{F}_p[x]/(x^n - 1)$$

be the polynomial corresponding to u. The polynomial corresponding to $(u_{n-1}, u_0, \ldots, u_{n-2})$ is $xf(x) \in \mathbf{F}[x]/(x^n - 1)$.

From now on we will usually identify a cyclic code of length n over $\mathbf{F}_p$ with the corresponding set of polynomials in $\mathbf{F}_p[x]/(x^n - 1)$. By Lemma 4.7, cyclic shifts of codewords correspond to muliplication by x.

Exercise: Let $C \subseteq \mathbf{F}_p[x]/(x^n - 1)$ be a cyclic code. Show that if $f(x) \in C$ and $h(x) \in \mathbf{F}_p/(x^n - 1)$ then $h(x)f(x) \in C$.

Generator polynomials

Definition 4.8

Let p be a prime. Let C be a cyclic code of length n over the finite field $\mathbf{F}_p$, identified with a subset of $\mathbf{F}_p[x]/(x^n - 1)$. A *generator polynomial* for C is a polynomial $g(x) \in \mathbf{F}_p[x]$ of degree $< n$ such that $g(x)$ divides $x^n - 1$ and

$$C = \left\{ \bar{f}(x)\bar{g}(x) : \bar{f}(x) \in \mathbf{F}[x]/(x^n - 1) \right\}.$$

Here a bar over a polynomial means that it should be considered as an element of $\mathbf{F}_p[x]/(x^n - 1)$. Thus the product $\bar{f}(x)\bar{g}(x)$ takes place in $\mathbf{F}_p[x]/(x^n - 1)$, not in $\mathbf{F}_p[x]$.

Example 4.9

Let $C = \{0, 1+x^2, x+x^3, 1+x+x^2+x^3\} \subseteq \mathbf{F}_2[x]/(x^4-1)$ be the polynomial version of the code in Example 6.2(2). We claim that $g(x) = 1+x^2$ is a generator polynomial for C.

Since $g(x)^2 = (1+x^2)^2 = 1+x^4 = x^4-1$, the polynomial $g(x)$ divides x^4-1. Every polynomial in C is a multiple of $1+x^2$.

Finally, suppose $\bar{f}(x) \in \mathbf{F}_2[x]/(x^4-1)$. Dividing $f(x)$ by $1+x^2$ we can write

$$f(x) = s(x)(1+x^2) + r(x)$$

where the degree of $r(x)$ is < 2. So $r(x) \in \{0, 1, x, 1+x\}$ and

$$f(x)(1+x^2) = s(x)(1+x^2)^2 + r(x)(1+x^2).$$

Hence, taking products in $\mathbf{F}_2[x]/(x^4-1)$ we have

$$\bar{f}(x)(1+x^2) = r(x)(1+x^2) \in C.$$

Generator polynomials

Exercise: Consider the code over $\mathbf{F}_3$ with codewords $\{(a, b, c, a, b, c) : a, b, c \in \mathbf{F}_3\}$. The corresponding subset of $\mathbf{F}_3[x]/(x^6 - 1)$ is

$$C = \{a + bx + cx^2 + ax^3 + bx^4 + cx^5 : a, b, c \in \mathbf{F}_3\}.$$

Find a generator polynomial for C.

Theorem 4.10

Let $\mathbf{F}$ be a finite field and let $C \subseteq \mathbf{F}[x]/(x^n - 1)$ be a cyclic code of length n. Then C has a generator polynomial.

Defining a Code Using a Generator Polynomial

Theorem 4.11

Let p be a prime, let $n \in \mathbf{N}$ and let $g(x) \in \mathbf{F}_p[x]/(x^n - 1)$ be a divisor of $x^n - 1$. If $g(x)$ has degree $r < n$ then

$$\{g(x), xg(x), \ldots, x^{n-r-1}g(x)\}.$$

is a basis for the cyclic code $C \subseteq \mathbf{F}_p[x]/(x^n - 1)$ with generator polynomial $g(x)$.

Proof (unfinished from last week): it is sufficient to show that the linear span of $\{g(x), xg(x), \ldots, x^{n-r-1}g(x)\}$ inside $\mathbf{F}_p[x]/(x^n - 1)$ is closed under multiplication by x. So it is enough to show that $x^{n-r}\bar{g}(x)$ is a linear combination of $g(x), xg(x), \ldots, x^{n-r-1}g(x)$.

Defining a Code Using a Generator Polynomial

Theorem 4.11

Let p be a prime, let $n \in \mathbf{N}$ and let $g(x) \in \mathbf{F}_p[x]/(x^n - 1)$ be a divisor of $x^n - 1$. If $g(x)$ has degree $r < n$ then

$$\{g(x), xg(x), \ldots, x^{n-r-1}g(x)\}.$$

is a basis for the cyclic code $C \subseteq \mathbf{F}_p[x]/(x^n - 1)$ with generator polynomial $g(x)$.

Proof (unfinished from last week): it is sufficient to show that the linear span of $\{g(x), xg(x), \ldots, x^{n-r-1}g(x)\}$ inside $\mathbf{F}_p[x]/(x^n - 1)$ is closed under multiplication by x. So it is enough to show that $x^{n-r}\bar{g}(x)$ is a linear combination of $g(x), xg(x), \ldots, x^{n-r-1}g(x)$.

Note that Theorem 4.11 shows that a cyclic code of length n with a generator polynomial of degree r over the finite field $\mathbf{F}_p$ has dimension $n - r$ and size p^{n-r}.

Binary Cyclic Codes of Length 7

Example 4.12

In $\mathbf{F}_2[x]$ we have

$$x^7 - 1 = (1+x)(1+x+x^3)(1+x^2+x^3)$$

where each factor is *irreducible*, i.e. the factors cannot be written as products of polynomials of small degree. The polynomial divisors of $x^7 - 1$ are therefore

$$1,\ 1+x,\ 1+x+x^3,\ 1+x^2+x^3,\ (1+x)(1+x+x^3),$$
$$(1+x)(1+x^2+x^3),\ (1+x+x^3)(1+x^2+x^3)$$

(1) The code with generator polynomial $1+x$ is the parity check extension of the code consisting of all binary words of length 6.

Example 4.12 [continued]

(2) Since

$$(1+x+x^3)(1+x^2+x^3) = 1+x+x^2+x^3+x^4+x^5+x^6$$

the code with this generator polynomial is the binary repetition code of length 7.

(3) The code with generator polynomial $1+x+x^3$ has generator matrix

$$\begin{pmatrix} 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 \end{pmatrix}$$

it is equivalent to the Hamming $[7,4,3]$-code. *Exercise:* prove this.

Generator matrices for Cyclic Codes

Theorem 4.13

Let C be a cyclic code of length n over $\mathbf{F}$ with generator polynomial $g(x) \in \mathbf{F}_p[x]$ of degree r. If $g(x) = a_0 + a_1 x + \cdots + a_r x^r$ then the $(n-r) \times n$ matrix

$$G = \begin{pmatrix} a_0 & a_1 & a_2 & \dots & a_r & 0 & \dots & 0 \\ 0 & a_0 & a_1 & \dots & a_{r-1} & a_r & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & a_0 & \dots & \dots & a_{r-1} & a_r \end{pmatrix}$$

is a generator matrix for C.

Encoding for Cyclic Codes

The encoding scheme on page 37 of the main notes would encode the number represented by $(b_0, b_1, \ldots, b_{n-r-1})$ in binary as

$$(b_0, b_1, \ldots, b_{n-r-1})G$$

As a polynomial, this codeword is

$$b_0 f(x) + b_1 x f(x) + \cdots + b_{n-r-1} x^{n-r-1} f(x) = (b_0 + b_1 x + \cdots + b_{n-r-1} x^{n-r-1}) f(x).$$

So we can encode messages in a cyclic code by polynomial multiplication. This can be performed more quickly than matrix multiplication.

Parity Check Matrices for Cyclic Codes

Theorem 4.14

Let C be a cyclic code over $\mathbf{F}_p$ of length n. Suppose that C has generator polynomial $g(x) \in \mathbf{F}_p$ of degree r and that g has distinct roots $c_1, \ldots, c_r \in \mathbf{F}_p$. Then the matrix

$$H = \begin{pmatrix} 1 & c_1 & c_1^2 & \cdots & c_1^{n-1} \\ 1 & c_2 & c_2^2 & \cdots & c_2^{n-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & c_r & c_r^2 & \cdots & c_r^{n-1} \end{pmatrix}$$

is a parity check matrix for C. The syndrome of a received word $v \in \mathbf{F}_p^n$ corresponding to the polynomial $k(x) \in \mathbf{F}_p[x]/(x^n - 1)$ is equal to

$$(k(c_1), \ldots, k(c_r)).$$